

# A numerically stable a posteriori error estimator for reduced basis approximations of elliptic equations

Andreas Buhr\*    Christian Engwer\*    Mario Ohlberger\*    Stephan Rave\*

March 17, 2014

## Abstract

The Reduced Basis (RB) method is a well established method for the model order reduction of problems formulated as parametrized partial differential equations. One crucial requirement for the application of RB schemes is the availability of an a posteriori error estimator to reliably estimate the error introduced by the reduction process. However, straightforward implementations of standard residual based estimators show poor numerical stability, rendering them unusable if high accuracy is required. In this work we propose a new algorithm based on representing the residual with respect to a dedicated orthonormal basis, which is both easy to implement and requires little additional computational overhead. A numerical example is given to demonstrate the performance of the proposed algorithm.

## 1 INTRODUCTION

Many problems in science and engineering require the solution of partial differential equations on large computational domains or very fine meshes. Even on modern hardware, standard discretization techniques for solving these problems can require many hours or even days of computation, which makes these approaches inapplicable for many-query situations like, e.g., design optimization, where the same problem has to be solved many times for different sets of parameters.

The Reduced Basis Method (RB) is by now a well-established tool for the model order reduction of problems formulated as parametrized partial differential equations. For a general introduction we refer to [1] and [2]. In an “offline phase”, a given high-dimensional discretization is solved for appropriately selected parameters and a reduced subspace is constructed as the span of these solution snapshots. In a later “online phase”, the problem can be solved efficiently for arbitrary new parameters via Galerkin projection onto the precomputed reduced space.

One crucial ingredient for the application of RB schemes is the availability of a quickly evaluable a posteriori error estimator to reliably estimate the error introduced by the reduction process. Such an estimator is also required by the weak greedy algorithm, which has been shown to be optimal for the generation of the reduced spaces [3], to efficiently perform an exhaustive search of the parameter space for parameters maximising the reduction error.

For affinely decomposed elliptic problems, a residual based error estimator is widely used [1, sec. 4.3]. In order to ensure quick evaluation of the dual norm of the residual, the computation is decomposed into high-dimensional operations during the “offline phase” and fast low-dimensional computations during the “online phase”. However, as observed by several authors [1, pp. 148–149][4][5], the implementation of this offline/online splitting shows poor numerical accuracy due to round-off errors which can render the estimator unusable when the given problem is badly conditioned and high accuracy is required. Observations suggest that the estimator typically stagnates at a relative error of order  $\sqrt{\varepsilon}$ , where  $\varepsilon$  is the machine accuracy of the floating point hardware used.

---

\*Institute for Computational und Applied Mathematics Einsteinstrasse 62, 48149 Münster, Germany, {andreas.buhr, christian.engwer, mario.ohlberger, stephan.rave}@uni-muenster.de

In the following, we propose a new algorithm to evaluate the norm of the residual which does not suffer the severe numerical problems of the traditional approach, is free of approximations, has only small computational overhead and is easy to implement.

To our knowledge, there is only one other contribution in which a numerically stable algorithm for evaluation of the estimator is presented [6, 7]. This approach however comes at the price of a computationally more expensive “online phase” (in [6]) or increased complexity of offline computations (in [7]) by application of the empirical interpolation method, which in turn requires additional stabilization. Moreover, a proof for the reliability of the modified estimator is missing in [7].

The remainder of this paper is organized as follows: In Section 2 we introduce the high-dimensional discrete problem that we will consider in this work. In Section 3 we summarize the Reduced Basis method including the weak greedy algorithm for basis generation. In Section 4 we present the residual based error estimator under consideration, the traditional algorithm for its evaluation as well as our proposed new algorithm. Finally, in Section 5 we give a numerical example underlining the improved stability of our new algorithm.

## 2 HIGH-DIMENSIONAL PROBLEM

We consider a discrete parametrized elliptic problem of the following form: let  $V$  be a Hilbert space of finite dimension  $N$ ,  $f_\mu \in V'$  a parametrized linear functional and  $a_\mu : V \times V \rightarrow \mathbb{R}$  a parametrized bilinear form such that for an  $\alpha_\mu > 0$  we have

$$\alpha_\mu \|\varphi\|_V^2 \leq a_\mu(\varphi, \varphi) \quad \forall \varphi \in V. \quad (1)$$

We then search for the solution  $u_\mu \in V$  satisfying

$$a_\mu(u_\mu, \varphi) = f_\mu(\varphi) \quad \forall \varphi \in V. \quad (2)$$

Note that the existence of a solution follows from the coercivity (1) of  $a_\mu$  and the finite dimensionality of  $V$ . The parameter  $\mu$  is confined to be an element of a fixed compact parameter space  $\mathcal{P} \subset \mathbb{R}^P$ . Moreover, we assume that  $a_\mu$  and  $f_\mu$  exhibit an affine parameter dependence, i.e. there exist parameter independent bilinear forms  $a^q : V \times V \rightarrow \mathbb{R}$  ( $1 \leq q \leq Q_a$ ), linear functionals  $f^q \in V'$  ( $1 \leq q \leq Q_f$ ) and coefficient functionals  $\theta_a^q : \mathcal{P} \rightarrow \mathbb{R}$  and  $\theta_f^q : \mathcal{P} \rightarrow \mathbb{R}$  such that

$$a_\mu(\varphi_1, \varphi_2) = \sum_{q=1}^{Q_a} \theta_a^q(\mu) a^q(\varphi_1, \varphi_2) \quad \text{and} \quad f_\mu(\varphi) = \sum_{q=1}^{Q_f} \theta_f^q(\mu) f^q(\varphi). \quad (3)$$

## 3 REDUCED BASIS APPROXIMATION

Given smooth dependence of the solution  $u_\mu$  on the parameter  $\mu$ , the dimension of the manifold of all solutions  $\{u_\mu \mid \mu \in \mathcal{P}\}$  is bounded by  $\dim(\mathcal{P})$  and, thus, is in general of much lower dimension than  $V$ . The Reduced Basis method exploits this fact by constructing a low-dimensional linear subspace  $\tilde{V} \subset V$  of dimension  $\tilde{N}$  in which the solution manifold can be approximated up to a small error. A reduced solution  $\tilde{u}_\mu \in \tilde{V}$  is then determined by Galerkin projection of (2) onto  $\tilde{V}$ , i.e. by solving

$$a_\mu(\tilde{u}_\mu, \tilde{\varphi}) = f_\mu(\tilde{\varphi}) \quad \forall \tilde{\varphi} \in \tilde{V}. \quad (4)$$

The solvability of (4) again follows from (1).

The reduced space  $\tilde{V}$  is constructed from the linear span of solutions to (2) for parameters selected by the following greedy search procedure: Starting with  $\tilde{V}_0 := \{0\} \subset V$ , in each iteration step the reduced problem (4) is solved and an error estimator is evaluated at all parameters  $\mu$  of a given training set  $\mathcal{S}_{train} \subset \mathcal{P}$ . If the maximum estimated error is below a prescribed tolerance  $tol$ , the algorithm stops. Otherwise, the high-dimensional problem (2) is solved for the parameter  $\mu_n^*$  maximising the estimated error and the reduced space is extended by the obtained solution snapshot:  $\tilde{V}_{n+1} := \tilde{V}_n \oplus \text{span}\{u_{\mu_n^*}\}$ .

## 4 RESIDUAL BASED A POSTERIORI ERROR ESTIMATOR

An a posteriori error estimator provides a computable upper bound for the model reduction error  $\|u_\mu - \tilde{u}_\mu\|_V$ . We consider here a widely used error estimator based on the discrete residual  $\mathcal{R}_\mu \in V'$  given by  $\mathcal{R}_\mu(\tilde{u}_\mu)(\varphi) := f_\mu(\varphi) - a_\mu(\tilde{u}_\mu, \varphi)$  for  $\varphi \in V$ .

**Theorem 4.1.** (Error bound) *The model reduction error  $\|u_\mu - \tilde{u}_\mu\|_V$  can be bounded using the dual norm of the residual and the coercivity constant of the bilinear form:*

$$\|u_\mu - \tilde{u}_\mu\|_V \leq \frac{1}{\alpha_\mu} \|\mathcal{R}_\mu(\tilde{u}_\mu)\|_{V'} \quad (5)$$

*Proof.* See [1, eq. 4.28].  $\square$

To calculate the dual norm of the residual  $\mathcal{R}_\mu(\tilde{u}_\mu)$  we make use of the fact that the norm of an element of  $\tilde{V}'$  is equal to the norm of its Riesz representative. Denoting by  $R : V' \rightarrow V$  the Riesz isomorphism and assuming the existence of a computable lower bound  $\alpha_{\mu, LB} \leq \alpha_\mu$  for the coercivity constant, we obtain a bound for the error containing only computable quantities:

$$\|u_\mu - \tilde{u}_\mu\|_V \leq \frac{1}{\alpha_{\mu, LB}} \|R(\mathcal{R}_\mu(\tilde{u}_\mu))\|_V \quad (6)$$

Direct evaluation of this error bound comprises the calculation of the Riesz representative and the computation of its norm, which are both high-dimensional operations. However, for the application of the RB method in many-query and real-time situations, it is crucial that the time for evaluating the a posteriori error estimator in the online phase is independent of the dimension of  $V$ . This is also required to make the use of large parameter training sets  $\mathcal{S}_{train}$  feasible, which is necessary to ensure optimal selection of the snapshot parameter  $\mu^*$ .

### 4.1 Traditional offline/online splitting

In order to avoid high-dimensional calculations during the online phase, the residual  $\mathcal{R}_\mu(\tilde{u}_\mu)$  can be rewritten using the affine decompositions (3) and a basis representation of  $\tilde{u}_\mu$ . Let  $\{\tilde{\psi}_1, \dots, \tilde{\psi}_{\tilde{N}}\}$  be a basis of  $\tilde{V}$  and let  $\tilde{u}_\mu = \sum_{i=1}^{\tilde{N}} \tilde{u}_{\mu_i} \tilde{\psi}_i$ , then the Riesz representative of the residual is given as

$$R(\mathcal{R}_\mu(\tilde{u}_\mu)) = \sum_{q=1}^{Q_f} \theta_f^q(\mu) R(f^q) - \sum_{q=1}^{Q_a} \sum_{i=1}^{\tilde{N}} \theta_a^q(\mu) \tilde{u}_{\mu_i} R(a^q(\tilde{\psi}_i, \cdot)). \quad (7)$$

To simplify notation, we rename the  $N_\eta := Q_f + Q_a \tilde{N}$  linear coefficients  $\theta_f^q(\mu)$  and  $\theta_a^q(\mu) \tilde{u}_{\mu_i}$  to  $\alpha_k$  and the vectors  $R(f^q)$  and  $R(a^q(\tilde{\psi}_i, \cdot))$  to  $\eta_k$ , i.e.  $R(\mathcal{R}_\mu(\tilde{u}_\mu)) = \sum_{k=1}^{N_\eta} \alpha_k \eta_k$ . The space  $\text{span}\{\eta_1, \dots, \eta_{N_\eta}\}$  is denoted by  $V_\eta$ . For the norm of the residual we obtain

$$\|R(\mathcal{R}_\mu(\tilde{u}_\mu))\|_V = \left( \sum_{k=1}^{N_\eta} \sum_{l=1}^{N_\eta} \alpha_k \alpha_l (\eta_k, \eta_l)_V \right)^{\frac{1}{2}}. \quad (8)$$

Using this representation, an offline/online decomposition of the error bound is possible by pre-computing the inner products  $(\eta_k, \eta_l)_V$  during the offline stage. In the online stage, only the sum in (8) has to be evaluated. As the number of summands is independent of the dimension of  $V$ , an online run-time independent of the dimension of  $V$  is achieved.

While this approach leads to an efficient computation of the residual norm, it shows poor numerical stability: in the sum (8), terms with a relative error of order of machine accuracy  $\varepsilon$  are added. Therefore, the sum shows an absolute error of at least  $\varepsilon$  times the largest value of  $|\alpha_k \alpha_l (\eta_k, \eta_l)_V|$ , and the error in the norm of the residual is thus at least of order  $\sqrt{\varepsilon} \cdot \sqrt{\max_{k,l} (|\alpha_k \alpha_l (\eta_k, \eta_l)_V|)}$ . This is in agreement with the observation that this algorithm stops converging at relative errors of order  $\sqrt{\varepsilon}$  (see Section 5).

---

**Algorithm 1:** Gram-Schmidt with re-iteration

---

**Input:** vectors  $v_i, i \in 1, \dots, N$   
**Output:** orthonormal vectors  $v_i$

```
1 for  $i \leftarrow 1, \dots, N$  do
2    $v_i \leftarrow v_i / \|v_i\|_V$ ;
3   repeat
4     for  $j \leftarrow 1, \dots, (i-1)$  do
5        $v_i \leftarrow v_i - (v_i, v_j)_V v_j$ ;
6     end
7     newnorm  $\leftarrow \|v_i\|_V$ ;
8      $v_i \leftarrow v_i / \text{newnorm}$ ;
9   until newnorm  $> 0.1$ ;
10 end
```

---

## 4.2 Improved offline/online splitting

While the floating point evaluation of (8) shows poor numerical accuracy, note that the evaluation of

$$\|\mathbf{R}(\mathcal{R}_\mu(\tilde{u}_\mu))\|_V = \left( \sum_{k=1}^{N_\eta} \alpha_k \eta_k, \sum_{k=1}^{N_\eta} \alpha_k \eta_k \right)_V^{\frac{1}{2}} \quad (9)$$

is numerically stable. Based on this observation, we propose a new algorithm to evaluate  $\|\mathbf{R}(\mathcal{R}_\mu(\tilde{u}_\mu))\|_V$  which is offline/online decomposable while maintaining the algorithmic structure of (9) to ensure stability.

The algorithm we propose evaluates (9) in the subspace  $V_\eta$  using an orthonormal basis for this space. It comprises three steps: 1. The construction of an orthonormal basis  $\Psi^\eta = \{\psi_1^\eta, \dots, \psi_{N_\eta}^\eta\}$  of  $V_\eta$ , 2. the evaluation of the basis coefficients of  $\eta_k$  w.r.t. the basis  $\Psi^\eta$  and 3. the evaluation of (9) using this basis representation. Note that this approach is offline/online decomposable: Steps 1 and 2 can be done offline, without knowing the parameter, while step 3 can be performed online. The size of the basis  $\Psi^\eta$  does not depend on the dimension of  $V$ .

In principle, any orthonormalization algorithm applied to  $\{\eta_k\}_{k=1}^{N_\eta}$  can be used for the computation of the basis  $\Psi^\eta$ . Note, however, that the algorithm has to compute the basis with very high numerical accuracy. As an example, the standard modified Gram-Schmidt algorithm usually fails to deliver the required accuracy. For the numerical example in Section 5, we have chosen an improved variant of the modified Gram-Schmidt algorithm, where vectors are re-orthonormalized until a sufficient accuracy is achieved (Algorithm 1).

After the basis  $\Psi^\eta$  has been constructed using an appropriate orthonormalization algorithm, we can compute for each  $\eta_k$  ( $1 \leq k \leq N_\eta$ ) basis representations  $\eta_k = \sum_{i=1}^{N_\eta} \bar{\eta}_{k,i} \psi_i^\eta$ , where  $\bar{\eta}_{k,i} = (\eta_k, \psi_i^\eta)_V$  due to the orthonormality of  $\Psi^\eta$ . The right-hand side of (9) can then be evaluated as:

$$\|\mathbf{R}(\mathcal{R}_\mu(\tilde{u}_\mu))\|_V = \left( \sum_{i=1}^{N_\eta} \left( \sum_{k=1}^{N_\eta} \alpha_k \bar{\eta}_{k,i} \right)^2 \right)^{\frac{1}{2}}, \quad (10)$$

which executes in time independent of the dimension of  $V$  and is observed to be numerically stable.

## 4.3 Run-time complexities

During the offline phase, both the traditional and the new algorithm have to calculate all Riesz representatives appearing in (7). This requires the application of the inverse of the inner product matrix for  $V$ , which can be computed in complexity  $\mathcal{O}(N \log(N))$  with appropriate preconditioners. As there are  $N_\eta$  Riesz representatives to be calculated, the overall run-time of this step is of order

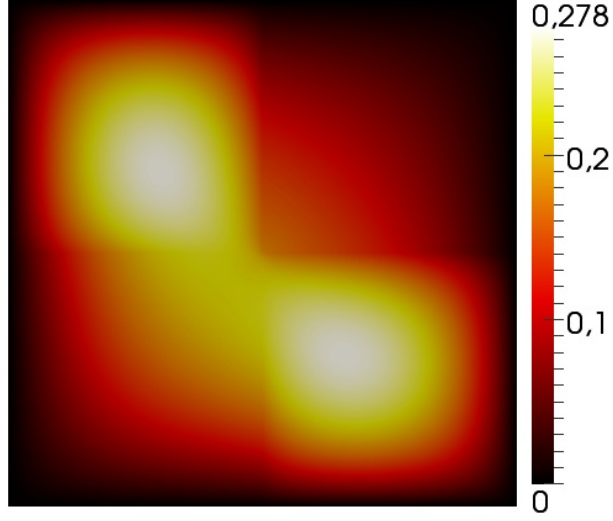


Figure 1: High-dimensional solution of (11) for  $\mu = (0.1, 1.0, 0.4, 1.0)$

$\mathcal{O}(N_\eta N \log(N))$ . The traditional algorithm proceeds with calculating all inner products  $(\eta_k, \eta_l)_V$  in (8), having a complexity of  $\mathcal{O}(N_\eta^2 N)$ . Thus the overall complexity of the offline phase for the traditional algorithm is  $\mathcal{O}(N_\eta^2 N + N_\eta N \log(N))$ .

After computing the Riesz representatives in (7), the improved algorithm generates the orthonormal basis  $\Psi^\eta$ . In practice it was observed that at most four re-iterations per vector are required during orthonormalization with Algorithm 1. Thus, choosing this algorithm for the generation of  $\Psi^\eta$  leads to a run-time complexity of  $\mathcal{O}(N_\eta^2 N)$  for this step. The calculation of the  $N_\eta^2$  basis coefficients  $\bar{\eta}_{k,i} = (\eta_k, \psi_i^\eta)_V$  has again complexity  $\mathcal{O}(N_\eta^2 N)$ , resulting in a total complexity of the offline phase for the new algorithm of  $\mathcal{O}(N_\eta^2 N + N_\eta N \log(N))$ , as for the traditional algorithm.

During the online phase, the right-hand sides of (8), resp. (10), are evaluated using the pre-computed quantities  $(\eta_k, \eta_l)_V$ , resp.  $\bar{\eta}_{k,i}$ . In both cases, a run-time of  $\mathcal{O}(N_\eta^2)$  is required.

All in all, both algorithms for evaluating (6) show the same run-time complexity, in the online phase as well as during the offline phase (Table 1). Note that  $\mathcal{O}(N_\eta^2) = \mathcal{O}(Q_f^2 + Q_a^2 \tilde{N}^2) = \mathcal{O}(\tilde{N}^2)$  for increasing reduced space dimensions.

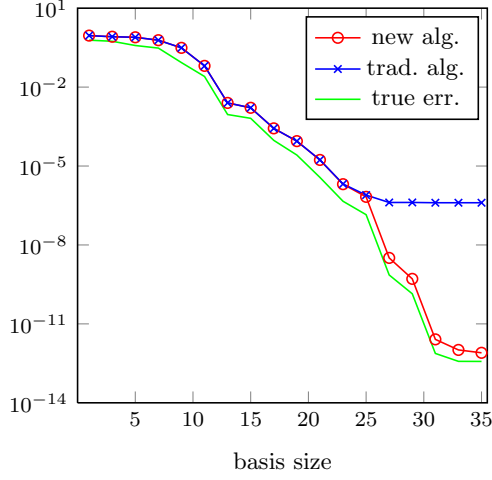
## 5 NUMERICAL RESULTS

In order to verify the improved numerical stability of our proposed algorithm, we considered an elliptic “thermal block” problem on the domain  $\Omega = [0, 1]^2$  of the form

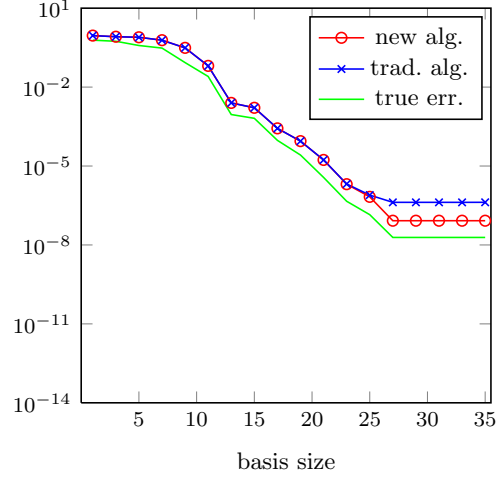
$$-\nabla \cdot (\sigma_\mu \nabla u_\mu) = 1, \quad u_\mu \in H_0^1(\Omega), \quad (11)$$

Table 1: Run-time complexities of traditional and new algorithm for evaluation of the error estimator.

stage	offline	online
traditional	$\mathcal{O}(N_\eta^2 N) + \mathcal{O}(N_\eta N \log(N))$	$\mathcal{O}(N_\eta^2)$
new	$\mathcal{O}(N_\eta^2 N) + \mathcal{O}(N_\eta^2 N) + \mathcal{O}(N_\eta N \log(N))$	$\mathcal{O}(N_\eta^2)$



(a) New algorithm used for basis generation



(b) Traditional algorithm used for basis generation

Figure 2: Maximum relative reduction errors and estimated reduction errors ( $H^1$ -norm) for numerical example (11).

with heat conductivity  $\sigma_\mu = \sum_{i,j=0}^1 \mu_{ij} \cdot \chi_{[i/2, (i+1)/2] \times [j/2, (j+1)/2]}$ , denoting by  $\chi_A$  the characteristic function of the set  $A$ . The parameters  $\mu = \{\mu_{ij}\}_{i,j=0}^1$  were allowed to vary in the space  $\mathcal{P} = [0.1, 1.0]^4$ .

Equation (11) was discretized using linear finite elements on a regular grid with  $500 \times 500 \times 2$  triangular entities (Fig. 1). Then, a reduced space of dimension 35 was generated with the weak greedy algorithm using our new algorithm for the evaluation of the error estimator. An equidistant training set of  $5^4$  parameters was used. Finally, for each  $n$ -dimensional reduced subspace  $\tilde{V}_n$  ( $0 \leq n \leq 35$ ) produced by the greedy algorithm we computed the maximum reduction error and the maximum estimated reduction errors using both the traditional and our improved algorithm on 20 randomly selected new parameters in  $\mathcal{P}$  (Fig. 2a). Moreover, the maximum and minimum efficiencies (i.e. the quotient error/estimate) of the estimator evaluated using both algorithms were determined for the same random parameters (Table 2). Our results clearly indicate the breakdown of the traditional algorithm for more than 25 basis vectors at a relative error of about  $10^{-7} \approx \sqrt{\varepsilon}$  whereas our new algorithm remains efficient for all tested basis sizes.

To underline the need for accurate error estimation in order to obtain reduced spaces of high approximation quality, we repeated the same experiment using the traditional algorithm for error estimation during basis generation (Fig. 2b). While the maximum model reduction error still improves from  $10^{-7}$  to  $10^{-8}$  after the breakdown of the error estimator, the final reduced space approximates the solution manifold 4 orders of magnitude worse than the space obtained with our improved algorithm.

## Acknowledgements

This work has been supported by the German Federal Ministry of Education and Research (BMBF) under contract number 05M13PMA and by CST - Computer Simulation Technology AG.

Table 2: Maximum and minimum efficiencies ( $H^1$ -norm) of traditional and new error estimator for numerical example (11); efficiencies were calculated for 20 randomly chosen parameters.

basis size		10	15	20	25	30	35
trad.	max	$4.9 \cdot 10^{-1}$	$4.3 \cdot 10^{-1}$	$4.6 \cdot 10^{-1}$	$4.1 \cdot 10^{-2}$	$1.9 \cdot 10^{-5}$	$9.0 \cdot 10^{-7}$
	min	$2.1 \cdot 10^{-1}$	$2.3 \cdot 10^{-1}$	$1.8 \cdot 10^{-1}$	$1.0 \cdot 10^{-3}$	$3.1 \cdot 10^{-6}$	$4.8 \cdot 10^{-7}$
new	max	$4.9 \cdot 10^{-1}$	$4.3 \cdot 10^{-1}$	$4.7 \cdot 10^{-1}$	$3.9 \cdot 10^{-1}$	$4.6 \cdot 10^{-1}$	$4.6 \cdot 10^{-1}$
	min	$2.1 \cdot 10^{-1}$	$2.3 \cdot 10^{-1}$	$2.2 \cdot 10^{-1}$	$2.1 \cdot 10^{-1}$	$2.3 \cdot 10^{-1}$	$2.4 \cdot 10^{-1}$

## References

- [1] A. T. Patera, G. Rozza. *Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations*, Version 1.0, Copyright MIT 2006, to appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering.
- [2] B. Haasdonk, M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *M2AN (Math. Model. Numer. Anal.)*, Vol. **42(2)**, 277–302, 2008.
- [3] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, P. Wojtaszczyk. Convergence Rates for Greedy Algorithms in Reduced Basis Methods. *SIAM J. Math. Anal.*, Vol. **43(3)**, 1457–1472, 2011.
- [4] M. Yano. A space-time Petrov-Galerkin certified reduced basis method: Application to the boussinesq equations. Accepted in *SIAM Journal on Scientific Computing*, 2013.
- [5] P. Benner, M. Hess. The Reduced Basis Method for Time-Harmonic Maxwell’s Equations. *Proceedings in Applied Mathematics and Mechanics*, Vol. **12**, 661–662, 2012.
- [6] F. Casenave. Accurate a posteriori error evaluation in the reduced basis method. *C. R. Math. Acad. Sci*, Vol. **350**, 539–542, 2012.
- [7] F. Casenave, A. Ern, T. Lelièvre. Accurate and online-efficient evaluation of the a posteriori error bound in the reduced basis method. Accepted in *M2AN (Math. Model. Numer. Anal.)*, 2013.